

محاضرة 6 : أساسيات ال intensity mapping

المحتوى : (1) تصنيف ال spatial processing

(2) أساسيات ال spatial processing

(3) ال intensity mapping

Dynamic Range stretch and threshold

Negative

Logarithmic

Power law trans.

(4) Piecewise Linear Transformation

(5) Intensity level and bit plane slicing

تصنيف ال spatial processing

* يمكن تقسيمها لخاصيتين

(1) Intensity Transformation : يرتبط فيها على بكسل واحد بسى ، ويمكنه تصغير التكامل

بأتنا تعاليم على neighborhood size يساوي 1×1

- من التطبيقات عليه هي ال Contrast Manipulation and image thresholding

* يمكن تقسيمه ثانية خاصة مراد ال spatial filtering

(2) Spatial Filtering : يشتغل على ال neighborhood rectangle ، يعني عندى

window ليها أبعاد $M \times N$ ، زي ما شغلنا فى ال blur ال محاضرة 4

- من التطبيقات عليه هي ال Contrast Manipulation and preprocessing

* لو فلتير ال window حجمها 1×1 بيت شغل Intensity transformation

أساسيات ال spatial processing

* عمليات ال spatial processing عادة بتكون Efficient وسهلة Resources

أقل للقارنة بال Transform operations

* الصيغة العامة هي

$$g(x,y) = T[f(x,y)]$$

- حيث $f(x,y)$ هي ال input image

- ال $g(x,y)$ هي ال output image

التي تعملها سواء على بكسل واحد

أو ال neighbors

لا تفكر في الصورة التي قولناه في الصورة 5 صنفه في 4 لو كان عندي pixels خارج حدود الصورة بسبب window عندي كل من اثنين:

- اما اني افرض قيم لل pixels (padding out of image pixels with a specific value)

- او اني اقلها لقيمة ال pixels دي نهائي

مثال على العملية T هي عملية averaging (قولناها قبل كده) التي بسبب blur

* عملية ال Transformation T مع ال Neighborhood Rectangle

(اللي هي ال window باي) بنسجها ال Spatial Filtering ممكن نسمي

ال neighborhood باحد اثنائه زي Rectangle

- Mask - Template - Kernel - Spatial Mask

* فكرت اني اقدر intensity mapping طارة طارة مع ال spatial transformation
 لما يكون ال rectangle ابعاده 1×1 (بشكل على البكسل نفسه وليس دعوة بار neighbors)

- في الحالة دي بتبقى المعادلة

$$S = T(r)$$

حيث r هي قيمة ال intensity في ال input

S هي قيمة ال pixel في ال output (نفس ال location)

T هو ال operator او العملية التي بعملها

* فكرت اني بنسج التطبيقات على ال image enhancement هو ال Image Enhancement

دوال ال intensity mapping

* الهدف الاساسي منها هو ال image enhancement

* ان image enhancement من مقرر على تحسين الصورة لـ human vision وحيث
 يمكن ان يكون غير واضح الصورة عنده نتيجه اوسع في حافة تصوير computing
 زي راني اوسع الصورة بحيث ندي recognition rate اعلى في تطبيق زي
 ال OCR

- في الحالة دي بيكونه المعيار عندي عند شكل الصورة اللي بتسوفه نكم ال recognition rate

* ما يبقى نكم نصف جودة وطريقة ال Enhancement **اعتماداً على حاجة**
 صم اتنين:

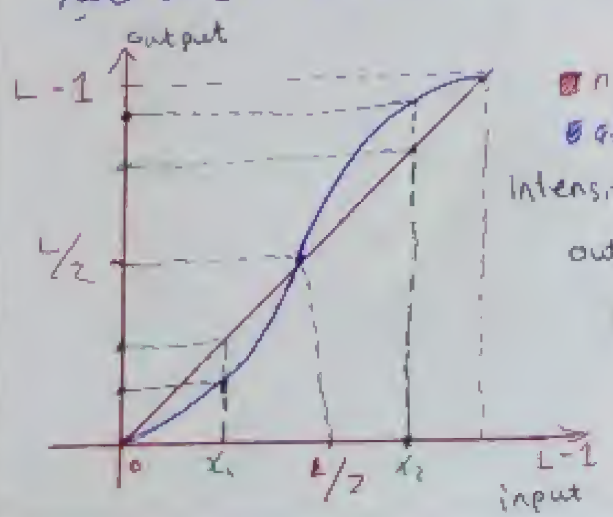
① في حالة تحسين الصورة لـ human investigation بيكونه اختيار
 طريقة ال Enhancement خاضع للـ subjective judgment اللي هي عمل
 ال investigation

② في حالة تحسين الصورة للـ Automated machine processing يبقى
 المعيار عندي objective زي ال recognition rate في ال OCR

بدال دوايل مشروحة كونس في ورق الستة اللي فاتت صفحة 25 و 26 و 27 و 28
 لترقيم البرنامج من ال صفحات

* هنوضح الفكرة ورا الرسم بتاع المعادلات عنده تفهم رازاي المعادلات
 بتعمل في ال Intensity وبيدها ستوية ملاحظات

* ابقى شوف ال Slides بالجرة مع الورد من اول سلايد [6.6]
 كد سلايد [6.17] هنلاقي خوية تطبيقات معاها وصور توضح أثر كل عملية



* لو فرضت محور أفقي عليه ال Intensity
 بتاعت ال input و المحور الراسي عليه
 ال Intensity بتاعت ال output ، في حالة واني مش هغير ال Intensity
 صيغة العلاقة بين ال output و ال input ، انه $output = input$
 و تدي علاقة linear بالأحمر
 - لو هبيت أقل ال بيكسلز ناحية ال أبيض (L-1) أفق
 و هبيت أكثر ال بيكسلز ناحية ال أسود (0)
 هعمل ال Mapping بالأزرق

[3]

• لنفرض لنظّم هيكلي بيك على كل ar functions

- $Thresholding$ لو وضع الـ $threshold$ أقل من K هتبقى أسود، و أكبر من K هتبقى أبيض، ده هيعمل الصورة لـ $Binary$ image

• تطبيقات كل $function$ موجودة في المحاضرات وورق أسنة اللي فانت، ركز وافيها كويس.

• في ar negative هتفكس قيم ar pixels، اللي كانت أبيض تخليك أسود و اللي كانت أسود تخليك أبيض.

• ركز على معرفة تطبيق كل $function$ ، ركز على معرفة تطبيق كل $function$

$Piecewise$ linear transformation
المفروض تكون في أكثر من الورق و ar slides لو فهمت زي ما قولتك في اللي فانت

$Intensity$ level slicing & bit-plane slicing
جزد ar $Intensity$ level slicing هشرح كويس في السلايد 6.17، و العور بتوضح التطبيق على كل $function$

الصورة (A) هي الصورة الأصلية، الصورة (B) هي تطبيق على ar $function$ اللي هي (a)، و الصورة (c) تطبيق على الدالة (b)

• نشرح ar bit-plane slicing بتبوية تفصيل كتير مش مترددة كويس، جازد slides

• ترقم ar planes و ar reconstruction للصورة فيه غلطة في ورف أسنة اللي فانت، فدها معهما والسلايد رقم [6.26]، [6.27]

• الفكرة هنا، انك عندي الصورة كل بكسل فيها قيمته بتتخزن في مجموعة من ar Bits، و بتقسم آفد أول Bit لكل ar pixels و أصل منك صورة $Binary$ ، و ثاني Bit آفده لكل ar pixels و أصل منك صورة $Binary$ وهكذا

• كل صورة بتسمى ar plane، لو عندي $K=8$ بيتقسم أصل ar 8 planes

* ترقيم ال planes يبدأ بـ [1] وينتهي بـ k ، بينما ترقيم ال Bits يبدأ من صفر وينتهي بـ $(k-1)$

يعني
 plane 1 \rightarrow hold bit - 0
 plane 2 \rightarrow hold bit - 1
 :
 plane $k \rightarrow$ hold bit $(k-1)$

* ال bits الصغيرة زي bit-0, bit-1, bit-2 تأثيرها في الصور مش كبير
 * ال bits الكبيرة زي bit-5, bit-6, bit-7 بيقت تأثيرها كبير ، وممكن
 أكتر في تطبيق زي ضغط الصورة (Compression) ، زاني أهمها قيم ال bits
 اللي هي (5, 6, 7) بس و أهمها الباقي (الصورة هتقل جودتها بس لحد
 مش صلاحت الفرق أوي)

* عشانم تشوف الصورة شكلها عامل رازاي في كل plane شوف سلايد
 [6.19]

* السبب ان ال bits الكبيرة أثرها أهم ، لانها بتدي قيم أكبر

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
↓	↓	↓	↓	↓	↓	↓	↓
0,128	0,64	0,32	0,16	0,8	0,4	0,2	0,1

هتدري ال bit رقم 7 مسؤولة لانها بتدي 128 أو صفر ، والرقم 128
 لما نضغط للصورة يدي تأثير أكبر بكثير من عدد ال bit-1 اللي بتدي [0, 2]
 * التطبيق هي سلايد 6 على ال Compression

* لما زاي أعمل ال bit-plane slicing ؟
 فيه طريقة سهلة (بس الدكتور ~~تسكتك~~ مش مقتنع بيها) أوي وكاين الطريقة
 اللي بفرسها (عند طريقه ال Masking زي ما كنا بنعمل في ال micro
 * لو عندنا قيمة بكنسل 11010111 = 215
 قانس أعمل AND مع كل bit وآخد ما لودها .

الطريقة التي عاينها الـكتور مختلفة شوية ، هنا قسم قيم الـ pixels لـ Ranges
 وصف الـ Range الذي عاينها الـ pixel بقول هو صورة الـ 1 في الـ plane

b_2	b_1	b_0	Value
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

هنا مخطط على صورتها $K=3$ للبيتم

plane 1 - bit 0

if value is odd then bit = 1

else then bit = 0

plane 2 - bit 1

if value $\in [0, 1]$ ← تنتمي الـ Range

or value $\in [4, 5]$

then bit = 0

if value $\in [2, 3]$

or value $\in [6, 7]$

then Bit = 1

plane 3 - bit 2

if value $\in [0, 3]$ then bit = 0

if value $\in [4, 7]$ then bit = 1

OR

if value ≥ 4 then bit = 1

else then bit = 0

* إزاي أرجع الصورة ؟

لضرب كل bit في الـ base بقاها و أجمعهم ، يعني لو فرضت عندي

صورة مختلطة بالـ planes رقم 6, 7, 8 و عاينها الـ pixel قيمته (الـ bits يعني)

في كل plane $p@plane 8=1$, $p@plane 7=1$, $p@plane 6=0$

Plane 8

Plane 7

Plane 6

$$P_{value} = 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5$$

$$= 128 + 64 + 0 = 192$$